Transactions on Computer Science and Intelligent Systems Research
ISSN: 2960-1800, eISSN: 2960-2238 | Volume 5
AIDML 2024

NeRF-based Real-Time Rendering Photo-Realistic Graphics Methods Review

Shunyu Yang

School of Computer Science, University of Glasgow, University Ave, Glasgow G12 8QQ, United Kingdom

100597@yzpc.edu.cn

Abstract. Recently, the publication of the Neural Radiance Field (NeRF) has sparked a surge in further research, unveiling a wealth of innovative solutions in related domains. With its potent predictive capabilities and operational efficiency, neural networks have addressed the traditional trade-off dilemma, enabling real-time rendering at state-of-the-art (SOTA) quality. This breakthrough has introduced a novel rendering approach in computer graphics (CG) and unlocked fresh opportunities for real-time applications such as video games, augmented reality (AR), and virtual reality (VR). This paper aims to offer a comprehensive overview of recent NeRF-like methodologies and explore potential pathways for enhancing NeRF to achieve both real-time performance and photorealistic standards. Our study includes a comparative analysis of these methodologies in terms of their efficacy and hardware requirements. Ultimately, this paper outlines potential future advancements in this field. The objective of this paper is to familiarize both newcomers and researchers with NeRF, catalyze in-depth investigations, propose enhanced methodologies.

Keywords: Neural Radiance Field, Computer Graphics, Neural Network Rendering, Real-Time.

1. Introduction

Before Artificial Intelligence (AI) was introduced to CG, many methods in CG could optimize the computer rendering to real-time and photo-realistic standard (here we after referenced real-time as at least 60 Frame Per Seconds). However, price, performance bottlenecks, visual effects, difficulty of implementation, cross-platform capabilities, and so on always need to be balanced. Even though great progress in Graphics Processing Units (GPUs) has been made compared to a few years ago, more theoretical support has been proposed such as Physically Based Rendering (PBR) and path tracing, there are still some users' accessibility is limited.

In the early development of AI, some non-real-time AI-aided rendering means have been proclaimed, which furnished CG with some AI references. These ideas exploited AI to improve the traditional rendering of distinct processes, however, performance is not good enough. Image-based Rendering (IBR) generates novel images of a scene from a set of pre-captured images. Generative Adversarial Networks (GANS) make use of a generator and a discriminator, two networks to create novel data that are indistinguishable from reality. Variational Autoencoders (VAEs) compress data to other representations, good for image generation and denoising. Combing PBR with Machine Learning (ML) to improve the quality and efficiency of complicated physical phenomena. Overall, these early attempts did not reach real-time and photo-realistic rendering at the same time, yet the above logic has played a very inspiring role in the current achievements.

Following the previous study, recent results show that neural network rendering can achieve both photo-realistic and real-time. Neural network, is known for powerful predictability, better integrated model data and led to a unique render approach, receiving an impressive profit on rendering. Neural network rendering, with a low correlation to traditional rendering, and possessing more AI features, can be considered a groundbreaking solution to CG rendering. It is immensely valuable to do further study on this.

NeRF was inspired by Structure from Motion (SFM) and Multi-view Stereo (MVS), first proposed positional encoding to basic neural network volume rendering, and offered it photo-realistic graphics as traditional rendering methods does, and made people start to regard neural networking as a new method of rendering [1]. However, the rendering method being used in the original NeRF cannot reach the real-time standard, although it attained the iconic achievement of AI-generated picture resolution. Within a few years, the author observed many sorts of strengthened NeRF were published, and some of them succeeded in rendering the 3D scene in real-time, these ideas raised neural network rendering to a higher state (Figure 1).

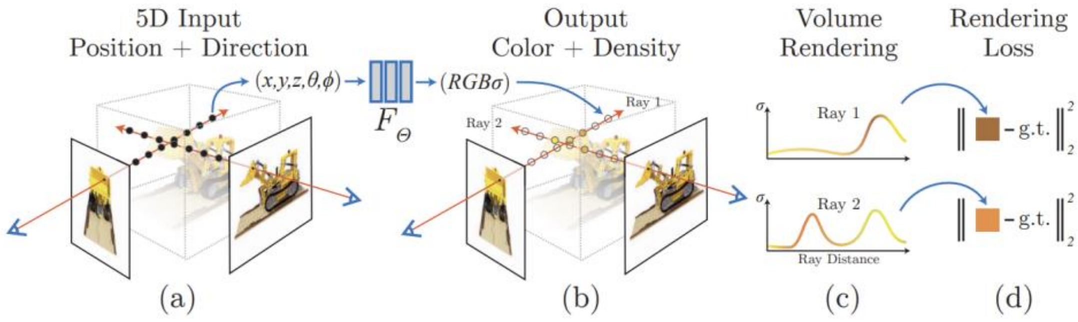

Fig.1 The NeRF volume rendering and training process [1]

The recent research on NeRF has developed so fast that many researchers have improved NeRF in many directions to many distinct extents, a lot of distracting content to people interested in it. The author intends to provide a brief but clear developing path on real-time rendering and filter out key measures and orientation to facilitate NeRF to both real-time and photo-realistic. For this, the paper has listed each idea initiated or improved NeRF to higher ground, compared each orientation step within each of them, clarified their properties, and summarized them to a categoric result. Overall, the author attempts to supply current real-time applications with a clearer optimizing direction, and promote consumer a better user experience.

Section 2 will detailed narrate advanced versions of NeRF as the development routine. 2.1, 2.2, and 2.3 will respectively depict three main directions, volume-based, surface-based, and hybrid methods. They all promote the NeRF reached real-time standards. Feature comparison will be described and shown in section 3. Some further latent upgrowth of NeRF will be displayed in Section 4, which depicts what NeRF-like methods could solve and what mishap NeRF-like methods have to comprehend (Figure 2).

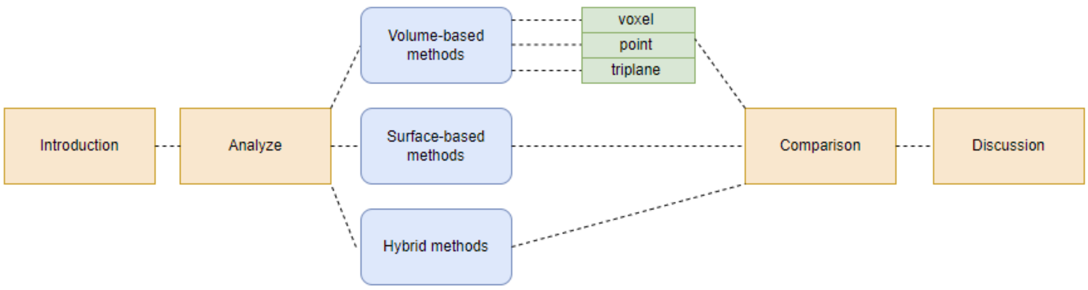

Fig. 2 Paper structure (Photo/Picture credit :Original)

2. Analyze

The Original NeRF proved the potent force of the implicit scene representation for a large scene, it can economize enormous storage by just building MLP instead of listing explicit data such as mesh, voxel, or point cloud directly in disk, and the synthetic image achieved the same HD as traditional explicit pipeline. However, after NeRF was published, researchers outcropped many potential problems and improved it to real-time standard. One main issue is NeRF still requires an expansive neural radiance field for color approximation so that is hard to index and access for later inference. Precisely, it equally and frequently builds the field along the camera ray for calculation efficiency, yet their sampling density directly decides the image resolution [2]. Not to mention extensive empty space is also encountered by fields and waste storage space.

To solve these problems, papers raised many ideas to make NeRF better. Various papers can be divided into three categories, simulating the volume in the scene as the radiance field, tracking and representing geometries' surface, and making use of both of them. At the beginning of NeRF development, volume-based methods are the foremost direction to escalate, which volumetrically render the scene by collecting the density and color in each field of view, as NeRF does. Some are aware the frequency of sampling becomes a huge storage burden and building surface features is enough and come up with surface construction. The hybrid noticed the surface took the loss of detailed geometric relations and tried to combine them. They are respectively narrated in Sections 2.1, 2.2 and 2.3.

2.1. Volume-based methods

The main problem volume-based methods encountered is how to describe the 3D geometry better than NeRF. Earlier approaches just solved the accessing speed. Some papers directly modified the storing data format, improving both indexing and accessing.

2.1.1. Voxel representation

An early optimization for NeRF, PlenOctree (Yu et al. 2021) noticed the problem of NeRF that numerous radiance fields lead to massive data accessing, which takes the majority of time-consuming inference. The solution mentioned is baking the trained neural network, saving it into octree, and using spherical harmonics to replace the polar coordinate in the original NeRF for easier reading [3]. PlenOctree can improve performance at least 3000 times on rendering when compared with the original NeRF. At the high cost of memory, in the best case [4], it can reach about 70FPS, which can be considered as the earliest method to meet both HD and real-time.

Afterward, another popular method, Sparse Neural Radiance Grid (SNeRG) (Hedman et al. 2021) is published. Similarly, it also transferred a trained Multi-Layer Perception (MLP) to Sparse Neural Radiance Grid [5], which solved complicated data structure results from frequent sampling of NeRF to some extent as well. Comparing the original NeRF and PlenOctree, the gird is more compact and reduced storage consumption. The following advancement's best performance reached 60 FPS [4], endowing the possibility of HD resolution at less cost of cache, which led to another welcomed early result.

2.1.2. Point representation

3DGS (Kerbl et al. 2023) came up with a unique geometry representation, a 3D Gaussian parameter-controlled ellipse, to simplify the data inquiring. Besides, this ellipse also contains the texture and density information. By iteratively adaptive modifying the parameters of the ellipse, its geometry can fulfill all models in the scene [6]. 3DGS reached both photo-realistic and real-time even better than current NeRF-like methods, which also shows data accessing weighed down the performance of NeRF a lot.

2.1.3. Triplane representation

After 3DGS came up with a competitive result on point-based and MLP-less way to render in real-time, SMERF (Streamable Memory Efficient Radiance Fields) (Duckworth et al. 2023) claimed they proved NeRF-like methods are still better than 3DGS [7]. SMERF is based on MERF (Reiser et al. 2023) uses a hash mesh compressed triplane data structure, and binarizes them to accelerate rendering [8]. Apart from MERF, the author announced that an extra memory is used to build sub-scenes and new distillation training to combine scenes up[6]. SMERF allows hardware-limited devices to run over 40 FPS stably, which 3DGS is unavailable, and its performance is slightly higher than 3DGS rendering on desktop (10 FPS).

2.2. Surface-based methods

The base idea of surface-based method is more fundamentally change the radiance field representation. The quality of synthesis image strongly related to massive radiance field NeRF required. By just simulating surface feature, surface-based methods saved more storage than volume-based methods.

A temporary performance peak is shown by MobileNeRF (Chen et al. 2022) in the early development. The team points out the problem of SNeRG's deferred rendering approach and its data format, sparse neural radiance grid, which cannot adapt to the common hardware. What's more, the cache SNeRG employed is still exceedingly high. MobileNeRF combines the NeRF and traditional render pipeline and takes the processed radiance field to a lightweight MLP in the fragment shader [9]. MobileNeRF successfully achieved real-time and platform adaptability, easily exceeding 60 FPS in the above situations. Phone for 10 FPS and desktop for 400 FPS still show the limitation of hardware though, in related areas this new benchmark is still universally compared as well.

BakedSDF (Yariv et al. 2023) found out that placing baked NeRF into a texture map as MobileNeRF directly is still unavailable and unrefined on most graphic processing applications, thus proposing an extended signed distance function representation. This pattern highly unified the triangle meshes format so that it is more available for large scenes and traditional applications [10] . In addition to device adaptability, the paper also provided data that tested as same condition as MobileNeRF and got 70 FPS rendering speed, about 10 FPS better than MobileNeRF.

The inference speed peak for surface-based means can be found in NeRF2Mesh (Tang et al. 2023) at 224 FPS in 1080p and 90 FPS in 4k resolution. Nerf2Mesh optimizes the quality of textured mesh extracted from NeRF and iteratively subdivides the surface to a simpler surface, which absorbs the advantage of MobileNeRF and BakedSDF and upgrade to a balanced and optimized version [11].

2.3. Hybrid methods

After surface-based methods were developed for enough time, some noticed the 3D detail is fractional sacrificed because of the property of surface representation. Partially involving volume features in surface-based methods can better balance efficiency and performance.

Hybrid methods trying to integrate other methods and VMesh (Guo et al. 2023) can be a successful instance. VMesh can reach 90 FPS in 4k resolution and 250 FPS in 1080p which is similar to NeRF2Mesh. In contrast, the representation Volume-Mesh used is capable of describing more detailed geometric structures. VMesh sequentially applied SDF representation, assigned newly proposed RefBasis texture on it, transforming to mesh to reduce the dependence on storage so that improved performance on consumer-grade devices [12]. The data shows the same FPS used as NeRF2Mesh yet almost a quarter of storage is used on the NeRF-Synthetic dataset, which is impressive about direct optimization without side effects.

Afterward, Binary Opacity Grids (BOG) (REISER et al. 2024) further released storage pressure by discrete opacity grid instead of continuous density field, averaged multiple rays to anti-alias, binarized the opacity value for swift accessing, and a fitting method for converting grid to mesh for adapting common render pipeline [13]. Unfortunately, there is no direct comparison between VMesh and BOG, the paper shows BOG with Temporal Anti-Alising (TAA) has almost doubled FPS than BakedSDF on desktops, and even three times on hardware-limited devices.

3. Comparison

In this section, former methods will be compared with numerical evidence and data in papers to summarize the strengths and shortcomings of each category. The performance this paper mentioned is based on a widely used dataset and running on the desktop. Details are shown in Table 1. FPS applied in Section 2 is based on Tanks & Temples (T&T)/unbounded 360° dataset if it is not mentioned. This dataset input and output 1920*1080 resolution images. Another mainly used resolution is 1600*1600. Although the MipNeRF-360 dataset did not provide detailed resolution, some data are provided based on it [14]. Table 2 illustrates the performance on the highest resolution on consumer-grade devices.

As Table 1 shows, the way how early methods reached real-time is paying high cost from different perspectives. PlenOctree had to pay over 1000 Megabytes and SNeRG sacrificed rendering speed even in relatively better equipment, the cost is huge to apply in industry applications. However, in terms of performance, later methods all have surpassed a lot than above two methods. According to the upsides and downsides of each device and varied resolution, VMesh and BOG are especially powerful when balancing FPS and storage. VMesh exceeds over 200 FPS at 10 MB storage, which can be an excellent solution to implicit neural networks in mobile applications. And BOG takes the position of cost-effective that doubled storage to exchange almost four times FPS. Hybrid methods show the best performance at present.

Table 1. Frame per seconds, storage, dataset, used device comparison. Data obtained from each paper, performance may vary, balanced best performance and same device as the author could find, for reference only.

	Plenoctree-1024 (PNG)	SNeRG (PN)	MobileNeRF	NERF2MESH	VMesh	BAKED SDF	BOG	3D GS	SMERF
FPS	67.1	52.8	98	231	250	412	927	260	278
Storage (MB)	1112.0	139.7	125.8	73.5	13.6	434.5	~200	740	153
Resolution/dataset	1920 x 1080		1600 x 1600			1920 x 1080		MipNeRF-360	
Device	Lenovo ThinkPad P1 Gen 2 notebook with a 4GB NVIDIA Quadro T1000		Macbook Pro (2020,M1) laptop			Desktop with NVIDIA RTX 3090		ThinkStation P620 with NVIDIA RTX 3090	

Some methods focused on supporting the capacity of cross-device, Table 2 demonstrated partial methods' performance in consumer-grade devices and performance in higher resolution. BOG is capable of different devices yet no detailed data is provided. The hybrid method still shows the merits that balancing from volume and surface allocates less storage, almost the same performance in all devices yet only a quarter of storage is used when comparing NeRF2Mesh and VMesh.

Table 2. Partial methods' 2k, 4k resolution supports and cross-device capacity

	iPhone13		iPad8		MacBookPro (2020, M1)			
	800	1600	800	1600	800	1600	2048	4096
SNeRG	-*	-*	-*	-*	43	15	9	2
MobileNeRF	57	-*	58	23	98	32	21	4
Nerf2Mesh	60*	60*	60*	60*	330	224	189	93
VMesh	60*	47	60	43	368	250	197	94

-*: Methods do not support on this device. 60*: The highest FPS on current device.

Depending on the published results in this paper, triplane indicates the best performance in volume representation, and hybrid methods performed the most outstanding one in all methods. The author speculates triplane is widely used in modern applications and hardware, more accelerating techniques are applied lead to this result. Hybrid methods are a trade-off and the latest solution, which thus can be the best temporary solution.

Besides, BOG made a simple comparison between volume, vertex, and triplane-voxel texture mapping, which is worthy to further prove the virtues of hybrid methods [12]. In the graph of BOG, vertex-based mapping took the least high memory consumption (VRAM), yet volume texture mapping displayed the best quality, finally, triplane-voxel has the highest FPS. As the result of BOG and this paper, the performance is highly relying on the storage and its accessing speed, which belongs to the hardware level. To render, NeRF developers should concern more about scene representations.

4. Discussion

This paper extensively discusses advanced versions of NeRF, including volume-based, surface-based, and hybrid methods, as part of their developmental trajectory. Through a deep exploration of these three main directions, this paper reveals the advantages and disadvantages of these methods in advancing the real-time capabilities of NeRF. Neural network rendering is getting closer to industry standards, and this part will make a summary of potential direction and trouble for future applications.

Firstly, traditional rendering heavily relies on explicit representation, leading to significant storage pressures for current 3D applications when constructing large-scale 3D scenes. NeRF-like methods can play a crucial role in alleviating this issue. Despite significant advancements in GPU technology enabling real-time applications on desktops or laptops at a relatively lower cost, achieving a comparable experience on mobile devices remains a challenge. For example, some mobile games have attempted to address this issue by utilizing the same model data for different characters, still resulting in substantial storage consumption exceeding 50 gigabytes. The emergence of NeRF demonstrates the potential of implicit representation, underscoring the ability of NeRF-like methods to achieve significant storage savings in large 3D scene. Consequently, storage constraints may no longer pose an obstacle in the future.

Secondly, currently, to enhance performance, NeRF developers must adhere to explicit standards across various applications and hardware, which can impede further advancements. While pure implicit methods may offer superior performance, they may not receive adequate support from devices or applications, thus failing to attract researchers' attention. Moreover, hardware-level adaptation may prove more effective than application-level modifications, as observed in prevalent methodologies. Some hardware manufacturers, including NVIDIA, have already begun exploring these possibilities. The author anticipates a rise in applications and hardware that support implicit methodologies, reducing the need for explicit representation adaptation and potentially leading to a significant performance enhancement.

Moreover, current neural network rendering lacks the flexibility for human intervention seen in traditional 3D editors, limiting artists' participation in dynamic scene generation. One viable solution involves integrating text-to-image techniques, wherein users input desired modifications to regulate Multilayer Perceptron (MLP) models in real-time, thereby altering the neural network to generate distinct scenes.

5. Conclusion

NeRF achieved remarkable outcomes in terms of photo-realistic image quality, innovative differential rendering, neural network, and providing traditional CG with a split-new approach for rendering. Apart from the publication of NeRF, the following advancements can be outstanding as well. One main lifting direction of NeRF is propelling the inference speed, which is also what the paper focused on. This topic has been developed in sufficient depth to reach real-time milestones. Three main approaches can all achieve real-time standards without massive losses of quality and eventually hybrid owns the best performance, which concludes the best scene representation based on former studies. The author also forecasted NeRF-like methods could solve some static scene generation and would have better performance with more hardware and application alterations. The most severe problem of NeRF is short of manual intervention to fill more artistic components and dynamic 3D. All in all, the NeRF-based method is a sorely promising direction to develop.

Reference

[1] Mildenhall B, Srinivasan P P, Tancik M, et al. Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM, 2021, 65(1): 99-106.

[2] Gao K, Gao Y, He H, et al. Nerf: Neural radiance field in 3d vision, a comprehensive review. arXiv preprint arXiv:2210.00379, 2022.

[3] Yu A, Li R, Tancik M, et al. Plenoctrees for real-time rendering of neural radiance fields, Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 5752-5761.

[4] Zhang J, Huang J, Cai B, et al. Digging into radiance grid for real-time view synthesis with detail preservation, European Conference on Computer Vision. Cham: Springer Nature Switzerland, 2022: 724-740.

[5] Hedman P, Srinivasan P P, Mildenhall B, et al. Baking neural radiance fields for real-time view synthesis, Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 5875-5884.

[6] Kerbl B, Kopanas G, Leimkühler T, et al. 3d gaussian splatting for real-time radiance field rendering. ACM Transactions on Graphics, 2023, 42(4): 1-14.

[7] Duckworth D, Hedman P, Reiser C, et al. SMERF: Streamable Memory Efficient Radiance Fields for Real-Time Large-Scene Exploration. arXiv preprint arXiv:2312.07541, 2023.

[8] Reiser C, Szeliski R, Verbin D, et al. Merf: Memory-efficient radiance fields for real-time view synthesis in unbounded scenes. ACM Transactions on Graphics, 2023, 42(4): 1-12.

[9] Chen Z, Funkhouser T, Hedman P, et al. Mobilenerf: Exploiting the polygon rasterization pipeline for efficient neural field rendering on mobile architectures, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023: 16569-16578.

[10] Yariv L, Hedman P, Reiser C, et al. Bakedsdf: Meshing neural sdfs for real-time view synthesis, ACM SIGGRAPH 2023 Conference Proceedings. 2023: 1-9.

[11] Tang J, Zhou H, Chen X, et al. Delicate textured mesh recovery from nerf via adaptive surface refinement, Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023: 17739-17749.

[12] Guo Y C, Cao Y P, Wang C, et al. VMesh: Hybrid volume-mesh representation for efficient view synthesis, SIGGRAPH Asia 2023 Conference Papers. 2023: 1-11.

[13] Reiser C, Garbin S, Srinivasan P P, et al. Binary Opacity Grids: Capturing Fine Geometric Detail for Mesh-Based View Synthesis. arXiv preprint arXiv:2402.12377, 2024.

[14] Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman.Mip-NeRF 360: Unbounded anti-aliased neural radiance fields.CVPR, 2022.